Package Tracker

Team Members:
Chase Grajeda(grajec)
Jianye Peng (pengj6)
Eddie Poon (poone)
Aneesh (koluka)

Feedback

- Missing setup instructions on GitHub Repository

AI

- Model is training
- Average accuracy currently sitting at around 42%
- Will keep adjusting hyperparameters and data augmentations
    - Epochs
    - Learning Rate
    - Lookback range
    - Momentum
    - Batch size
    - Decay rate

Blockchain

- Blockchain component is nearly finished
- Need to integrate it with AI component
- We already have the code to launch the smart contracts.

Integration:

- Created an API for the compiled smart contract
- Allow for frontend to interact with smart contract

```python
#Change this to post
#Update Status of package
@app.get("/status/{status}")
def updateStatus(status: str):
    global nonce
    nonce += 1
    #Update status of package
    package_tracker_root = w3.eth.contract(address=transaction_receipt.contractAddress, abi=abi)
    previous_status = package_tracker_root.functions.displayStatus().call()
    update_status = package_tracker_root.functions.updateStatus(status).build_transaction({
    "chainId": chain_id, "from": address, "gasPrice": w3.eth.gas_price, "nonce": nonce
    })
    #Sign the transaction
    sign_update = w3.eth.account.sign_transaction(
        update_status, private_key = private_key
    )
    #Send the transaction
    send_start = w3.eth.send_raw_transaction(sign_update.rawTransaction)
    w3.eth.wait_for_transaction_receipt(send_start)
    new_status = package_tracker_root.functions.displayStatus().call()
    return { "Package was successfully updated" if (previous_status != new_status) else "Package was not updated"}

#Get status of package
@app.get("/details/")
def getDetails():
    package_tracker_root = w3.eth.contract(address=transaction_receipt.contractAddress, abi=abi)
    return {package_tracker_root.functions.displayStatus().call()}
```

Demo/Frontend:

Welcome to Package Tracker

Add a Package | Claim a Package | View Details of a Package

Q & A